■ ประพนธ์ อัศวภาณุวัฒน์

SOFT MAIL

Query & Report
ตอนที่ 2

กล่าวถึงส่วนการใช้งาน Database Query เป็นหลัก โดยจะกล่าวถึงฟอร์มต่าง ๆ ที่ใช้งาน Query ในโปรแกรม

สวัสดีครับ ในฉบับนี้เรายังคุยกันเรื่อง Phone Reporter Project ต่อจากฉบับที่แล้ว สำหรับเนื้อหาจะกล่าวถึงส่วนการใช้งาน Database Query เป็นหลัก โดยจะกล่าวถึงฟอร์มต่าง ๆ ที่ใช้งาน Query ในโปรแกรม เพื่อไม่ให้เสียเวลาเราเริ่มกันเลยครับ

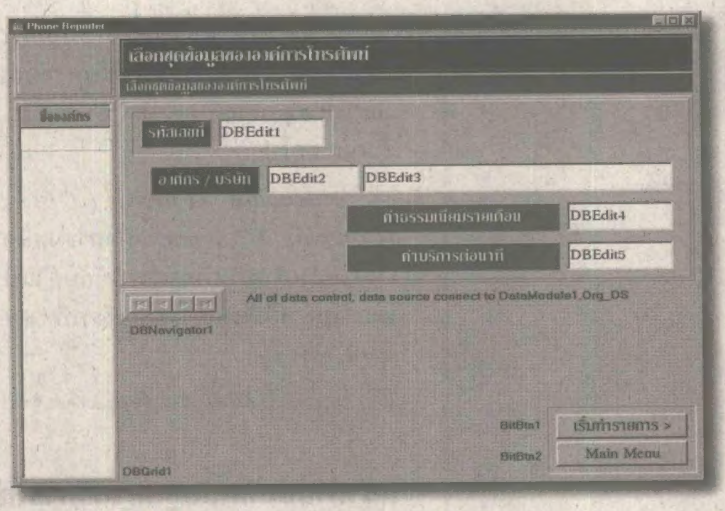
การใช้งาน Query แบ่งได้เป็น 2 ส่วนคือ การค้นหาข้อมูล (Retrive Data) และการปรับปรุงข้อมูล (Update Data) ในส่วนการค้นหาข้อมูลเราสามารถใช้เพียง Tquery Object ในการดำเนินงาน โดยกำหนด SQL Statement ให้กับ SQL Property ดังนี้

```
with Datamodule1.Org_Query do
begin
    Close; // สั่งให้ยกเลิกหรือปิด connection เดิม
    SQL.Clear;  // ลบชุดคำสั่งเดิมที่มีอยู่ใน SQL Command List
    // บรรจุคำสั่ง SQL Statement ลงใน SQL Command List
    SQL.Add( 'Select * from Organization_Table' );
    Open;  // สั่งให้เริ่มต้นติดต่อกับ database เพื่อค้นหาข้อมูล
end;
```

สำหรับการปรับปรุงข้อมูล เราสามารถใช้ TUpdateSQL Object ช่วยในการดำเนินงานในส่วนนี้ โดยกำหนด Property ให้กับ DataModule1.Org_Query.CachedUpdates เท่ากับ True ในกรณีที่ใช้การปรับปรุงข้อมูลแบบ Cached Updates ซึ่งเป็นวิธีที่ผมแนะนำให้ใช้ และให้ DataModule1.Org_Query.Update Object เท่ากับ DataModule1.Org_Update SQL; เพื่อเป็นการกำหนดให้ส่วน Update Object ของ Org_Query ใช้ออบเจ็กต์ Update SQL ตามด้วยการระบุชุดคำสั่ง SQL Statement สำหรับดำเนินการในส่วนของกิจกรรม ลบข้อมูล เพิ่มข้อมูล และแก้ไขข้อมูล ไว้ใน DeleteSQL, InsertSQL และ ModifySQL ของ Datamoduel1.Org_UpdateSQL ตามลำดับ

เมื่อดำเนินการใช้คำสั่งปรับปรุงข้อมูลเรียบร้อย จะต้องใช้คำสั่ง ApplyUpdate เพื่อเป็นการโอนย้ายคำสั่งและชุดข้อมูลจาก Internal Cache ไปปรับปรุงยังฐานข้อมูลจริง ลักษณะการใช้งานต่าง ๆ ขอให้ติดตามจากฟอร์มที่จะกล่าวถึงดังต่อไปนี้ครับ

Delphi



รูปที่ 1
ฟอร์ม FITEM1_1
เลือกองค์กร/บริษัท เพื่อทำรายการ

หน้าต่างเลือกองค์กร (Fitem1_1/UITEM1_1.PAS)

ฟอร์ม FITEM1_1 แสดงอยู่ในรูปที่ 1 และมีส่วนของโปรแกรมแสดงอยู่ในลิสติงที่ 1 ใช้สำหรับเลือกองค์กร/บริษัทที่ให้บริการด้านการโทรศัพท์ เพื่อทำการบันทึกข้อมูลตามใบแจ้งหนี้ที่ได้รับ (เนื่องจากเวลารับใบแจ้งหนี้จะรับเป็นชุด ๆ ละหลาย ๆ ใบจากองค์กร/หรือบริษัทที่ให้บริการ) มีส่วนของโปรแกรมที่สำคัญดังต่อไปนี้

ลิสติงที่ 1 โปรแกรม UITEM1_1.PAS

```
00001: // Phone Reporter
00002: // List 2.1 : UITEM1_1.PAS
00003: //
00004: unit uitem1_1;
00005: interface
00006: uses
00007:   Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
00008:   StdCtrls, Buttons, ExtCtrls, DBCtrls, Mask, Grids, DBGrids, DBCGrids;
00009: type
00010:   TFitem1_1 = class(TForm)
00011:     Bevel1: TBevel;
00012:     BitBtn1: TBitBtn;
00013:     BitBtn2: TBitBtn;
00014:     DBGrid1: TDBGrid;
00015:     Bevel2: TBevel;
00016:     Panel1: TPanel;
00017:     Panel2: TPanel;
00018:     Panel3: TPanel;
00019:     Panel4: TPanel;
00020:     DBNavigator1: TDBNavigator;
00021:     DBEdit1: TDBEdit;
00022:     DBEdit2: TDBEdit;
00023:     Panel5: TPanel;
00024:     DBEdit3: TDBEdit;
00025:     DBEdit4: TDBEdit;
00026:     DBEdit5: TDBEdit;
00027:     Panel6: TPanel;
00028:     Panel7: TPanel;
00029:     Panel8: TPanel;
00030:     Bevel3: TBevel;
00031:     PmonthlySelect: TPanel;
00032:     procedure FormCreate(Sender: TObject);
00033:     procedure FormCloseQuery(Sender: TObject; var CanClose: Boolean);
00034:     procedure BitBtn1Click(Sender: TObject);
00035:     procedure BitBtn2Click(Sender: TObject);
00036:     procedure DBGrid1CellClick(Column: TColumn);
00037:   private
00038:     { Private declarations }
00039:     procedure InitVar;
00040:     procedure ReleaseVar;
00041:     procedure RetriveOrganization;
00042:   public
00043:     { Public declarations }
00044:   end;
00045: var
00046:   Fitem1_1: TFitem1_1;
```

Delphi

```
00047:  implementation
00048:  uses uprDMdle, uPrMenu, uPrMisc, uItem1_2;
00049:  {$R *.DFM}
00050:  //-------------------------------------------------
00051:  procedure TFitem1_1.InitVar;
00052:  begin
00053:    PmonthlySelect.caption:=' ทำรายการของเดือน '+GetSelectMonth;
00054:    RetriveOrganization;
00055:  end;
00056:  procedure TFitem1_1.ReleaseVar;
00057:  begin
00058:    Fitem1_1.release; Fitem1_1:=nil;
00059:  end;
00060:  //-------------------------------------------------
00061:  procedure TFitem1_1.RetriveOrganization;
00062:  begin
00063:    with Datamodule1.Org_Query do
00064:    begin
00065:      Close;
00066:      SQL.Clear;
00067:      SQL.Add( 'Select * from Organization_Table' );
00068:      Open;
00069:    end;
00070:  end;
00071:  //-------------------------------------------------
00072:  procedure TFitem1_1.FormCreate(Sender: TObject);
00073:  begin
00074:    InitVar;
00075:  end;
00076:  procedure TFitem1_1.FormCloseQuery(Sender: TObject; var CanClose: Boolean);
00077:  begin
00078:    ReleaseVar;
00079:    Canclose:=true;
00080:  end;
00081:  procedure TFitem1_1.BitBtn1Click(Sender: TObject);
00082:  begin
00083:    if ( curSelOrg='' ) then
00084:    begin
00085:      ShowMessage( 'กรุณาเลือกชื่อองค์กร/บริษัท เพื่อทำรายการต่อไป' ); exit;
00086:    end;
00087:    if ( Fitem1_2=nil ) then
00088:      Application.CreateForm( TFItem1_2, Fitem1_2 );
00089:    Fitem1_2.Show; Fitem1_1.close;
00090:  end;
00091:  procedure TFitem1_1.BitBtn2Click(Sender: TObject);
00092:  begin
00093:    FprMenu.Show; Fitem1_1.close;
00094:  end;
00095:  procedure TFitem1_1.DBGrid1CellClick(Column: TColumn);
00096:  var s : shortstring;
00097:  begin
00098:    s:=column.Grid.SelectedField.Text;
00099:    if ( s<>'' ) then
00100:    begin
00101:      curSelOrg:=s; curSelOrgNum:=DBEdit1.Field.AsInteger;
00102:    end;
00103:  end;
00104:  end.
```

procedure TFitem1_1.RetriveOrganization; กิจกรรมสำหรับอ่านข้อมูลของ Organization_Table ทั้งหมด เข้าสู่ dataset ด้วยคำสั่ง SQL Statement ดังต่อไปนี้

```
'Select * from Organization_Table'
```

procedure TFitem1_1.BitBtn1Click(Sender: TObject); กิจกรรมสำหรับตรวจสอบค่าตัวแปร curSelOrg และเปิดฟอร์ม Fiem1_2 เพื่อทำงานในขั้นตอนต่อไป

procedure TFitem1_1.DBGrid1CellClick (Column: TColumn); กิจกรรมชนิด Event ของ DBGrid1 สำหรับกำหนดค่าให้กับตัวแปร curoSelOrg โดยอ่านค่าที่ได้จากการคลิกเลือกที่ DBGrid1 ของผู้ใช้ด้วยคำสั่ง

```
Column.Grid.SelectedField.Text
```

หน้าต่างบันทึกข้อมูลค่าบริการโทรศัพท์ (Fitem1_2/UITEM1_2.PAS)

ฟอร์ม Fitem1_2 แสดงอยู่ในรูปที่ 2 และมีส่วนของโปรแกรมแสดงอยู่ในลิสติงที่ 2 เป็นฟอร์มสำหรับบันทึกข้อมูลค่าใช้โทรศัพท์เข้า database โดยจะบันทึกตามเงื่อนไขที่เลือกจากฟอร์มก่อนหน้านี้ มีกิจกรรมที่สำคัญดังต่อไปนี้

procedure TFitem1_2.RetrivePhoneNumber; กิจกรรมสำหรับอ่านหมายเลขโทรศัพท์ทั้งหมด ที่ตรงกับรหัสผู้ให้บริการ เพื่อให้ผู้ใช้เลือกหมายเลขโทรศัพท์ ด้วยคำสั่ง

```
'Select * from telephone_table where orgCode=x'
```

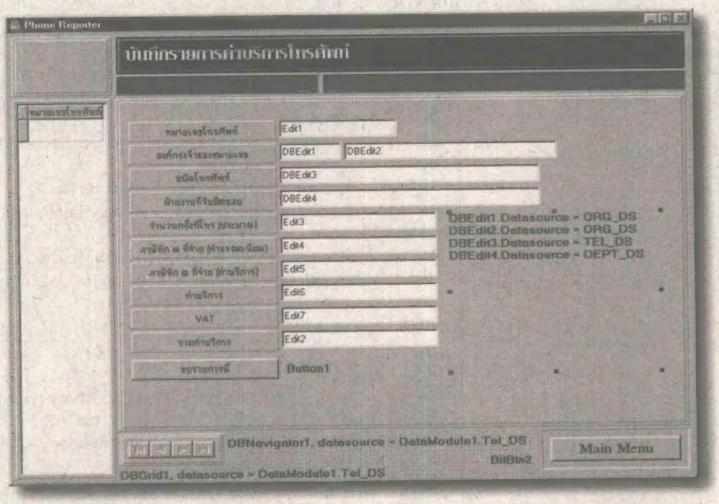
procedure TFitem1_2.ApplyUpdate; กิจกรรมสำหรับบันทึกข้อมูลจาก Internal Cache เข้าสู่ Database ที่ติดต่ออยู่ด้วยชุดคำสั่ง

```
with DataModule1.Trans_Query do
begin
  DataModule1.Database1.StartTransaction;
 try
   ApplyUpdates;
   DataModule1.Database1.Commit;
 except
   DataModule1.Database1.Rollback;
 raise;
 end;
 commitUpdates;
end;
```

procedure TFitem1_2.CalVAT_TAX (totAmount : double); กิจกรรมคำนวณหาค่าภาษีมูลค่าเพิ่ม โดยแสดงการอ่านค่าโดยตรงจาก dataset ของ Table ต่าง ๆ ณ เรคอร์ดปัจจุบัน คำนวณค่าภาษีหัก ณ ที่จ่าย และค่าภาษีมูลค่าเพิ่ม แล้วบันทึกรายการลง Transaction_Table ด้วยคำสั่ง

Delphi

```
monthlyFee:=DataModule1.Org_DS.DataSet.FieldByname( 'Monthly_Fee' ).AsFloat; //ค่าบริการรายเดือน
servRatePerMin:=DataModule1.Org_DS.DataSet.FieldByname( 'Serv_Rate_Per_Min' ).AsFloat; // ค่าบริการต่อนาที
feeTAX:=DataModule1.Org_DS.DataSet.FieldByname( 'Fee_TAX' ).AsFloat; //อัตราภาษีหัก ณ ที่จ่ายค่าธรรมเนียมบริการ
callTAX:=DataModule1.Org_DS.DataSet.FieldByname( 'call_TAX' ).AsFloat; //อัตราภาษีหัก ณ ที่จ่ายค่าบริการโทรศัพท์
VATrate:=DataModule1.Org_DS.DataSet.FieldByname( 'VAT_Rate' ).AsFloat; //อัตราภาษีมูลค่าเพิ่ม
telVAT:=GetVATValue( totAmount, VATrate ); //ถอด VAT ออกจากค่าบริการรวม
chargeExcludeVAT:= totAmount - telVAT; //คำนวณหาค่าบริการไม่รวม VAT
telCallCount:=integer( round( ( chargeExcludeVAT - monthlyFee ) / servRatePerMin ) ); // คำนวณหาจำนวนครั้งที่โทร
telFeeTAXamount:=( monthlyFee * feeTAX ) / 100;  //คำนวณหา FeeTAX (ถ้ามี)
telChargeTAXamount:=(( chargeExcludeVAT - monthlyFee ) * callTAX) / 100;  //คำนวณหา CallTAX (ถ้ามี)
```



รูปที่ 2
ฟอร์ม FITEM1_2
หน้าต่างบันทึกรายการ
ค่าบริการโทรศัพท์

ลิสติงที่ 2
โปรแกรม UITEM1_2.PAS

```
00001: // Phone Reporter
00002: // List 2.2 : UITEM1_2.PAS
00003: //
00004: unit uitem1_2;
00005: interface
00006: uses
00007:   Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
00008:   ExtCtrls, StdCtrls, Buttons, Grids, DBGrids, DBCtrls, Mask, db;
00009: type
00010:   TFitem1_2 = class(TForm)
00011:     Bevel1: TBevel;
00012:     BitBtn2: TBitBtn;
00013:     Panel2: TPanel;
00014:     Panel1: TPanel;
00015:     PmonthlySelect: TPanel;
00016:     Panel3: TPanel;
00017:     Panel4: TPanel;
00018:     DBNavigator1: TDBNavigator;
00019:     Bevel2: TBevel;
00020:     DBGrid1: TDBGrid;
00021:     Panel5: TPanel;
00022:     Panel6: TPanel;
00023:     Panel7: TPanel;
00024:     Panel8: TPanel;
00025:     Panel9: TPanel;
00026:     Panel10: TPanel;
00027:     Panel11: TPanel;
00028:     Panel12: TPanel;
00029:     Panel13: TPanel;
00030:     Panel14: TPanel;
00031:     DBEdit2: TDBEdit;
00032:     DBEdit3: TDBEdit;
00033:     DBEdit4: TDBEdit;
00034:     Edit1: TEdit;
00035:     DBEdit1: TDBEdit;
00036:     Edit2: TEdit;
00037:     Edit3: TEdit;
00038:     Edit4: TEdit;
00039:     Edit5: TEdit;
00040:     Edit6: TEdit;
00041:     Edit7: TEdit;
00042:     Porg: TPanel;
00043:     Button1: TButton;
00044:     procedure FormCreate(Sender: TObject);
00045:     procedure FormCloseQuery(Sender: TObject; var CanClose: Boolean);
00046:     procedure BitBtn2Click(Sender: TObject);
00047:     procedure Edit1KeyUp(Sender: TObject; var Key: Word;
00048:       Shift: TShiftState);
00049:     procedure DBGrid1CellClick(Column: TColumn);
00050:     procedure DBGrid1KeyUp(Sender: TObject; var Key: Word;
00051:       Shift: TShiftState);
00052:     procedure Edit2KeyUp(Sender: TObject; var Key: Word;
00053:       Shift: TShiftState);
00054:     procedure DBGrid1ColEnter(Sender: TObject);
00055:     procedure Button1Click(Sender: TObject);
00056:   private
00057:     { Private declarations }
00058:       trans_Type : integer;
00059:     procedure InitVar;
00060:     procedure ReleaseVar;
00061:     procedure ClearDisplayTextField;
00062:
00063:     procedure ApplyUpdate;
00064:     procedure CalVAT_TAX( totAmount : double );
00065:     procedure SearchTransaction( telId : string );
00066:
```

Delphi

```
00067:      procedure RetrivePhoneNumber;
00068:    public
00069:      { Public declarations }
00070:    end;
00071:  var
00072:    Fitem1_2: TFitem1_2;
00073:  implementation
00074:  uses uPrMenu, uPrMisc, uprDMdle;
00075:  {$R *.DFM}
00076:  //---------------------------------------------------
00077:  procedure TFitem1_2.RetrivePhoneNumber;
00078:  begin
00079:    with DataModule1.Tel_Query do
00080:    begin
00081:      close;
00082:      SQL.Clear;
00083:      SQL.add( 'select * from telephone_table' );
00084:      SQL.add( 'where' );
00085:      SQL.add( '  OrgCode = '+IntToStr( curSelOrgNum ) );
00086:      open;
00087:    end;
00088:  end;
00089:  //---------------------------------------------------
00090:  procedure TFitem1_2.InitVar;
00091:  begin
00092:    trans_Type:=NOT_SELECT;
00093:    PmonthlySelect.caption:='  ทำรายการของเดือน '+GetSelectMonth;
00094:    Porg.caption:=' '+curSelOrg;
00095:    RetrivePhoneNumber;
00096:  end;
00097:  procedure TFitem1_2.ReleaseVar;
00098:  begin
00099:    Fitem1_2.release; Fitem1_2:=nil;
00100:  end;
00101:  procedure TFitem1_2.ClearDisplayTextField;
00102:  begin
00103:    Edit2.text:=''; Edit3.text:=''; Edit4.text:='';
00104:    Edit5.text:=''; Edit6.text:=''; Edit7.text:='';
00105:  end;
00106:  //---------------------------------------------------
00107:  procedure TFitem1_2.ApplyUpdate;
00108:  begin
00109:    with DataModule1.Trans_Query do
00110:    begin
00111:      DataModule1.Database1.StartTransaction;
00112:      try
00113:        ApplyUpdates;
00114:        DataModule1.Database1.Commit;
00115:      except
00116:        DataModule1.Database1.Rollback;
00117:      raise;
00118:      end;
00119:      commitUpdates;
00120:    end;
00121:  end;
00122:  procedure TFitem1_2.CalVAT_TAX( totAmount : double );
00123:  var monthlyFee  : double;
00124:      chargeExcludeVAT : double;
00125:      feeTax, callTAX : double;
00126:      VATrate, telVAT : double;
00127:      servRatePerMin : double;
00128:      tmp : double;
00129:      tmpS : shortString; p : pchar;
00130:      telChargeTaxAmount : double;
00131:      telCallCount : integer;
00132:      telServCharge : double;
00133:      telFeeTAXamount : double;
00134:      lastTransRec : longint;
00135:      deptCode : integer;
00136:  begin
00137:  telServCharge:=0; telCallCount:=0; telFeeTAXamount:=0;
        telChargeTAXamount:=0;
00138:  monthlyFee:=DataModule1.Org_DS.DataSet.FieldByname
        ( 'Monthly_Fee' ).AsFloat;
00139:  servRatePerMin:=DataModule1.Org_DS.DataSet.FieldByname
        ( 'Serv_Rate_Per_Min' ).AsFloat;
00140:   // get fee TAX rate
00141:   feeTAX:=DataModule1.Org_DS.DataSet.FieldByname( 'Fee_TAX' ).AsFloat;
00142:  // get service TAX rate
00143:   callTAX:=DataModule1.Org_DS.DataSet.FieldByname( 'call_TAX' ).AsFloat;
00144:  // get VAT rate
00145:   VATrate:=DataModule1.Org_DS.DataSet.FieldByname( 'VAT_Rate' ).AsFloat;
00146:  if ( servRatePerMin=0 ) then
00147:  begin
00148:  ShowMessage( 'ค่าบริการต่อนาทีผิด กรุณาตรวจสอบข้อมูลอีกครั้ง
        [Tel_Organization.Serv_Rate_Per_Min]' );
00149:    exit;
00150:  end;
00151:  // decompose VAT value
00152:  telVAT:=GetVATValue( totAmount, VATrate );
00153:  chargeExcludeVAT:= totAmount - telVAT;
00154:  // extract call count
00155:  telCallCount:=integer( round( ( chargeExcludeVAT - monthlyFee ) servRatePerMin ) );
00156:  // is valid call count ?
00157:  tmp:=( chargeExcludeVAT-monthlyFee );
00158:  if ( round( tmp ) mod round( servRatePerMin )) <> 0 then
00159:  begin
00160:  ShowMessage( ' คำเตือน : ไม่สามารถหาจำนวนการโทรที่แน่นอนจาก จำนวนเงิน
        " & Tel_Charge_Amount & " บาท, ระบบจะทำการบันทึกข้อมูลให้
        แต่กรุณาตรวจสอบให้แน่นอนอีกครั้ง ' );
00161:  //   exit;
00162:  end;
00163:  if ( feeTAX>0 ) and ( callTAX>0 ) then
00164:  begin // in case TOT, TA do not have return TAX
00165:  telFeeTAXamount:=( monthlyFee * feeTAX ) / 100;  //Get fee tax amount
00166:  telChargeTAXamount:=(( chargeExcludeVAT - monthlyFee ) * callTAX) / 100;
        //Get charge tax amount
00167:  end;
00168:  telServCharge:=chargeExcludeVAT;
00169:
00170:
00171:  Edit3.text:=IntToStr( telCallCount );
00172:  Edit4.text:=FloatToStrF( telFeeTAXamount, ffnumber, 12, 2 );
00173:  Edit5.text:=FloatToStrF( telChargeTAXamount, ffnumber, 12, 2 );
00174:  Edit6.text:=FloatToStrF( telServCharge, ffnumber, 12, 2 );
00175:  Edit7.text:=FloatToStrF( telVAT, ffnumber, 12, 2 );
00176:
00177:  with DataModule1.Trans_Query do
00178:  begin
00179:    FieldByName( 'Month' ).asInteger:=curSelMonth+1;
00180:    FieldByName( 'Year' ).asInteger:=curYear;
00181:    FieldByName( 'RecDate' ).asDateTime:=Date;
00182:    FieldByName( 'TelNo' ).asString:=Edit1.text;
00183:     DeptCode:=DataModule1.Dept_Query.FieldByName( 'Dept_No' ).AsInteger;
00184:    FieldByName( 'Dept_No' ).AsInteger:=DeptCode;
00185:    FieldByName( 'ServCharge' ).AsFloat:=telServCharge;
00186:    FieldByName( 'VAT' ).AsFloat:=telVAT;
00187:    FieldByName( 'TotAmount' ).AsFloat:=totAmount;
00188:    FieldByName( 'DialCount' ).AsInteger:=telCallCount;
00189:    FieldByName( 'FeeTAX' ).AsFloat:=telFeeTAXamount;
00190:    FieldByName( 'ServTAX' ).AsFloat:=telChargeTAXamount;
00191:  end;
00192:  ApplyUpdate;
00193:  trans_Type:=NOT_SELECT;
00194:  //  DataModule1.Trans_Query.ApplyUpdates;
00195:  end;
00196:  procedure TFitem1_2.SearchTransaction( telId : string );
00197:  var ftmp : double; itmp, deptCode : longint;
00198:  begin
00199:    deptCode:=DataModule1.Tel_Query.FieldByName( 'Dept_No' ).asInteger;
00200:    DataModule1.SearchDeptName( deptCode );
00201:    with DataModule1.Trans_Query do
00202:    begin
00203:      Close;
```

Delphi

```
00204:     SQL.Clear;
00205:     SQL.Add( 'Select * from Transaction_Tot_Table' );
00206:     SQL.Add( 'where' );
00207:     SQL.Add( '  Month = '+IntToStr( curSelMonth+1 ) );
00208:     SQL.Add( 'and' );
00209:     SQL.Add( '  TelNo = '+#39+telId+#39 );
00210:     SQL.Add( 'and' );
00211:     SQL.Add( '  Year = '+IntToStr( curYear ) );
00212:     Open;
00213:   end;
00214:   if ( DataModule1.Trans_Query.IsEmpty ) then
00215:   begin
00216:     ClearDisplayTextField;
00217:     trans_Type:=INSERT_TYPE; DataModule1.Trans_Query.Insert;
00218:   end else
00219:   begin
00220:     // retrive previous data
00221:     ftmp:=DataModule1.Trans_Query.FieldByName( 'TotAmount' ).AsFloat;
00222:     Edit2.text:=Format('%8.2f', [ftmp]);
00223:     ltmp:=DataModule1.Trans_Query.FieldByName( 'DialCount' ).AsInteger;
00224:     Edit3.text:=IntToStr( ltmp );
00225:     ftmp:=DataModule1.Trans_Query.FieldByName( 'FeeTax' ).AsFloat;
00226:     Edit4.text:=Format('%8.2f', [ftmp]);
00227:     ftmp:=DataModule1.Trans_Query.FieldByName( 'ServTax' ).AsFloat;
00228:     Edit5.text:=Format('%8.2f', [ftmp]);
00229:     ftmp:=DataModule1.Trans_Query.FieldByName( 'ServCharge' ).AsFloat;
00230:     Edit6.text:=Format('%8.2f', [ftmp]);
00231:     ftmp:=DataModule1.Trans_Query.FieldByName( 'VAT' ).AsFloat;
00232:     Edit7.text:=Format('%8.2f', [ftmp]);
00233:     trans_Type:=EDIT_TYPE; DataModule1.Trans_Query.Edit;
00234:   end;
00235: end;
00236: //-------------------------------------------------
00237: procedure TFitem1_2.FormCreate(Sender: TObject);
00238: begin
00239:   InitVar;
00240:   Edit1.text:=DBGrid1.Columns.Grid.SelectedField.AsString;
00241: end;
00242: procedure TFitem1_2.FormCloseQuery(Sender: TObject; var CanClose: Boolean);
00243: begin
00244:   ReleaseVar;
00245:   CanClose:=true;
00246: end;
00247: procedure TFitem1_2.BitBtn2Click(Sender: TObject);
00248: begin
00249:   FPrMenu.Show; Fitem1_2.close;
00250: end;
00251: procedure TFitem1_2.Edit1KeyUp(Sender: TObject; var Key: Word;
00252:   Shift: TShiftState);
00253: var telNo : ansiString;
00254: begin
00255:   case key of
00256:     VK_RETURN :
00257:     begin
00258:       telNo:=Edit1.text;
00259: if ( not DataModule1.Tel_Query.Locate('Tel_No',telNo,[ loCaseInsensitive ]))then
00260:       begin
00261:         Edit1.text:='';
00262:         ShowMessage( 'ไม่พบ หมายเลขโทรศัพท์ '+telNo+' ในระบบ' ); exit;
00263:       end;
00264:       SearchTransaction( Edit1.text );
00265:       Application.ProcessMessages;
00266:       Edit2.SetFocus; Edit2.SelectAll;
00267:     end;
00268:     VK_UP :
00269:     begin
00270:       DataModule1.Tel_Query.Prior;
00271:       Edit1.text:=DBGrid1.SelectedField.AsString;
00272:       SearchTransaction( Edit1.text );
00273:     end;
00274:     VK_DOWN :
00275:     begin
00276:       DataModule1.Tel_Query.Next;
00277:       Edit1.text:=DBGrid1.SelectedField.AsString;
00278:       SearchTransaction( Edit1.text );
00279:     end;
00280:   end;
00281: end;
00282: procedure TFitem1_2.DBGrid1CellClick(Column: TColumn);
00283: begin
00284:   Edit1.text:=Column.Grid.SelectedField.AsString;
00285:   SearchTransaction( Edit1.text );
00286:   Edit1.Setfocus;
00287: end;
00288: procedure TFitem1_2.DBGrid1KeyUp(Sender: TObject; var Key: Word;
00289:   Shift: TShiftState);
00290: begin
00291:   Edit1.text:=DBGrid1.Columns.Grid.SelectedField.AsString;
00292: end;
00293: procedure TFitem1_2.Edit2KeyUp(Sender: TObject; var Key: Word;
00294:   Shift: TShiftState);
00295: var totAmount : double; code : integer;
00296: begin
00297:   case key of
00298:     VK_RETURN :
00299:     begin
00300:       if ( Edit1.text='' ) then exit;
00301:       val( Edit2.text, totAmount, code );
00302:       if ( code<>0 ) then
00303:       begin
00304:         ShowMessage( 'Invalid total amount' );
00305:         Edit2.SetFocus; Edit2.SelectAll;
00306:         exit;
00307:       end;
00308:       // in case, customer re-edit ( correct ) data
00309:       if ( trans_Type=NOT_SELECT ) then
00310:       begin
00311:         SearchTransaction( Edit1.text );
00312:         CalVAT_TAX( totAmount );
00313:         SearchTransaction( Edit1.text );
00314:       end else CalVAT_TAX( totAmount );
00315:       Edit1.setfocus;
00316:     end;
00317:     VK_UP, VK_DOWN :
00318:     begin
00319:       Edit1.setfocus; edit1.selectAll;
00320:     end;
00321:   end;
00322: end;
00323: procedure TFitem1_2.DBGrid1ColEnter(Sender: TObject);
00324: begin
00325:   Edit1.text:=DBGrid1.Columns.Grid.SelectedField.AsString;
00326:   SearchTransaction( Edit1.text );
00327:   Edit1.SetFocus;
00328: end;
00329: procedure TFitem1_2.Button1Click(Sender: TObject);
00330: var ret : word;
00331: begin
00332: ret:=MessageDlg( 'Delete '+Edit1.text+', are you sure ?', mtConfirmation,
00333:                  mbOKcancel, 0 );
00334:   if ( ret=mrOK ) then
00335:   begin
00336:     trans_Type:=DELETE_TYPE;
00337:     DataModule1.Trans_Query.Delete;
00338:     Application.ProcessMessages;
00339:     ApplyUpdate;
00340:     SearchTransaction( Edit1.text );
00341:   end;
00342: end;
00343: end.
```

Delphi

procedure TFitem1_2.SearchTransaction(telId : string); กิจกรรมสำหรับตรวจสอบรายการค่าบริการการใช้โทรศัพท์ ด้วยคำสั่ง

```
'Select * From Transaction_TOT_Table where
( Month = x ) and ( TelNo = y ) and ( Year = z )'
```

เพื่อตรวจสอบว่ารายการนี้เคยทำการบันทึกข้อมูลหรือไม่ โดยค้นหาด้วยข้อมูลของเดือน ปี และหมายเลขโทรศัพท์ ที่จะทำการบันทึก เมื่อรัน Query แล้วจะตรวจสอบ Result ของข้อมูลด้วยคำสั่ง DataModule 1.Trans_Query.IsEmpty ถ้าเป็น TRUE แสดงว่ายังไม่มีรายการนี้ในฐานข้อมูล เราจะกำหนดให้โปรแกรมเข้าสู่สถานะ INSERT MODE เพื่อทำการเพิ่มข้อมูลเข้าสู่ฐานข้อมูลในระบบ และในกรณีที่เป็น FALSE หมายถึงมีข้อมูลชุดนี้อยู่ในระบบ ก็จะทำการ Retrive Data สำหรับ Record ที่ค้นพบ และปรับสถานะ Query เป็น MODIFY MODE เพื่อทำการแก้ไขข้อมูล

procedure TFitem1_2.Edit1KeyUp(Sender: TObject; var Key: Word; กิจกรรมชนิด Event สำหรับรับหมายเลขโทรศัพท์ และทำการตรวจสอบว่ามีอยู่ในระบบหรือไม่

procedure TFitem1_2.Edit2KeyUp(Sender: TObject; var Key: Word; กิจกรรมชนิด Event สำหรับรับค่าบริการรวม VAT เพื่อนำไปคำนวณต่อด้วยกิจกรรมอื่นอีกครั้ง

procedure TFitem1_2.Button1Click(Sender: TObject); กิจกรรมชนิด Event สำหรับลบ record ปัจจุบันออกจากฐานข้อมูลด้วยคำสั่ง

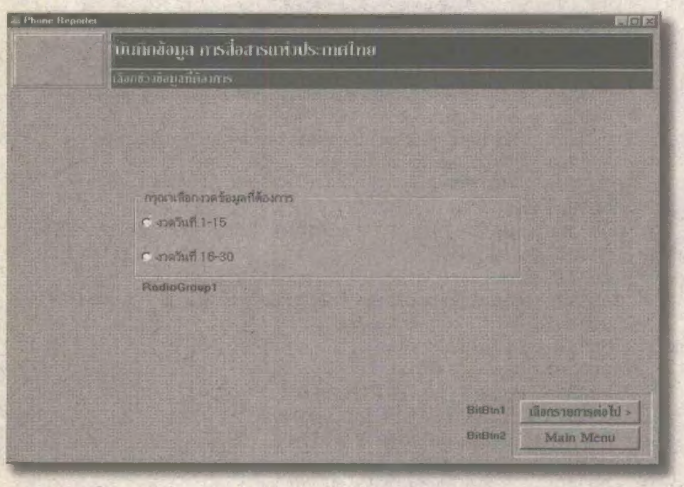
```
DataModule1.Trans_Query.Delete; //ลบข้อมูล record ปัจจุบัน
Application.ProcessMessages; //ให้ application ดำเนินการคำสั่งภายใน
ApplyUpdate; // ปรับปรุงข้อมูลจริง
```

สำหรับการคำนวณอัตราภาษีหัก ณ ที่จ่าย แบ่งแยกคำนวณเป็นสองส่วนคือ อัตราภาษีหัก ณ ที่จ่ายสำหรับค่าธรรมเนียมรายเดือน และอัตราภาษีหัก ณ ที่จ่ายสำหรับค่าบริการรายเดือน การคำนวณภาษีหัก ณ ที่จ่ายจะคำนวณเฉพาะบริษัทเอกชนที่ให้บริการโทรศัพท์ และจะนำยอดภาษีนี้หักออกจากค่าบริการและค่าธรรมเนียมก่อนทำการจ่ายยอดค่าใช้จ่ายจริงให้กับบริษัทผู้ให้บริการโทรศัพท์

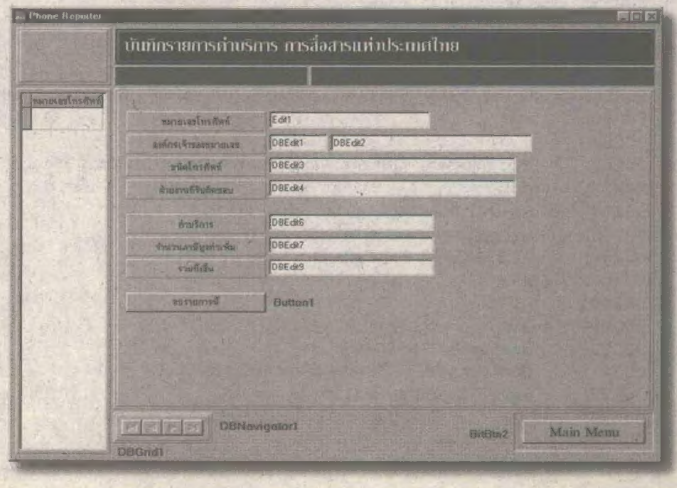
หน้าต่างเลือกช่วงเวลา (งวด) ที่บันทึก (Fitem2_1/UITEM2_1.PAS)

ฟอร์ม Fitem2_1 แสดงดังรูปที่ 3 เป็นฟอร์มสำหรับเลือกช่วงเวลาที่ต้องการบันทึก (งวดที่ 1 หรืองวดที่ 2) โดยอ่านค่าจาก RadioGroup แล้วให้ค่ากับตัวแปร curPeriod ซึ่งเป็นตัวแปรชนิด Global นำไปใช้ในฟอร์มถัดไป

สำหรับฟอร์มนี้เป็นฟอร์มหนึ่งที่มีเนื้อหาไม่ซับซ้อน ผมขออนุญาตกล่าวถึงเฉพาะรูปแบบฟอร์มและหน้าที่ในการใช้งานเท่านั้น โดยไม่ขอกล่าวถึงส่วนซอร์สโค้ดดังที่กล่าวไว้แล้วในบทความชุดแรกครับ



รูปที่ 3
ฟอร์ม FITEM2_1 สำหรับเลือกช่วงเวลา (งวดในการบันทึกข้อมูล)



รูปที่ 4
ฟอร์ม FITEM2_2
หน้าต่างบันทึกค่าบริการ การสื่อสาร

Delphi

ลิสติงที่ 3 โปรแกรม UITEM2_2.PAS

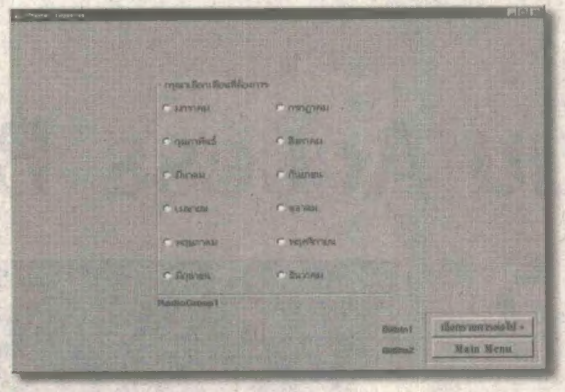
```
00001: // Phone Reporter
00002: // List2.3 : UITEM2_2.PAS
00003: //
00004: unit uItem2_2;
00005: interface
00006: uses
00007:   Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
00008:   ExtCtrls, StdCtrls, Buttons, DBCtrls, Grids, DBGrids, Mask, db;
00009: type
00010:   TFitem2_2 = class(TForm)
00011:     Bevel1: TBevel;
00012:     BitBtn2: TBitBtn;
00013:     Panel2: TPanel;
00014:     Panel1: TPanel;
00015:     PmonthlySelect: TPanel;
00016:     Pperiod: TPanel;
00017:     Bevel2: TBevel;
00018:     DBGrid1: TDBGrid;
00019:     Panel4: TPanel;
00020:     DBNavigator1: TDBNavigator;
00021:     Panel3: TPanel;
00022:     Panel5: TPanel;
00023:     Panel6: TPanel;
00024:     Panel7: TPanel;
00025:     Panel8: TPanel;
00026:     Panel10: TPanel;
00027:     Panel11: TPanel;
00028:     Panel13: TPanel;
00029:     Button1: TButton;
00030:     Edit1: TEdit;
00031:     DBEdit1: TDBEdit;
00032:     DBEdit2: TDBEdit;
00033:     DBEdit3: TDBEdit;
00034:     DBEdit4: TDBEdit;
00035:     DBEdit6: TDBEdit;
00036:     DBEdit7: TDBEdit;
00037:     DBEdit9: TDBEdit;
00038:     Label1: TLabel;
00039:     Label2: TLabel;
00040:     Label3: TLabel;
00041:     Label4: TLabel;
00042:     procedure FormCreate(Sender: TObject);
00043:     procedure FormCloseQuery(Sender: TObject; var CanClose: Boolean);
00044:     procedure BitBtn2Click(Sender: TObject);
00045:     procedure DBGrid1CellClick(Column: TColumn);
00046:     procedure DBGrid1ColEnter(Sender: TObject);
00047:     procedure DBGrid1KeyUp(Sender: TObject; var Key: Word;
00048:       Shift: TShiftState);
00049:     procedure Edit1KeyUp(Sender: TObject; var Key: Word;
00050:       Shift: TShiftState);
00051:     procedure DBEdit6KeyUp(Sender: TObject; var Key: Word;
00052:       Shift: TShiftState);
00053:     procedure Button1Click(Sender: TObject);
00054:   private
00055:     { Private declarations }
00056:       trans_Type : integer;
00057:     procedure InitVar;
00058:     procedure ReleaseVar;
00059:     procedure RetrivePhoneNumber;
00060:     procedure RetriveOrgName( orgCode : integer );
00061:     procedure SearchTransaction( telId : string );
00062:     procedure ApplyUpdate;
00063:     procedure CalVAT_TAX( serviceCharge : double );
00064:   public
00065:     { Public declarations }
00066:   end;
00067: var
00068:   Fitem2_2: TFitem2_2;
00069: implementation
00070: uses uprDMdle, uPrMenu, uPrMisc;
00071: {$R *.DFM}
00072: //---------------------------------------------------
00073: procedure TFitem2_2.RetrivePhoneNumber;
00074: begin
00075:   with DataModule1.Tel_Query do
00076:   begin
00077:     close;
00078:     SQL.Clear;
00079:     SQL.add( 'select * from telephone_table' );
00080:     open;
00081:   end;
00082: end;
00083: procedure TFitem2_2.RetriveOrgName( orgCode : integer );
00084: begin
00085:   with DataModule1.Org_Query do
00086:   begin
00087:     close;
00088:     SQL.Clear;
00089:     SQL.Add( 'Select * from Organization_Table' );
00090:     SQL.add( 'where' );
00091:     SQL.add( '  orgCode = '+IntToStr( orgCode ) );
00092:     open;
00093:   end;
00094: end;
00095: procedure TFitem2_2.SearchTransaction( telId : string );
00096: var ftmp : double; ltmp, deptCode, orgCode : longint;
00097: begin
00098:   deptCode:=DataModule1.Tel_Query.FieldByName( 'Dept_No' ).asInteger;
00099:   orgCode:=DataModule1.Tel_Query.FieldByName( 'OrgCode' ).asInteger;
00100:   DataModule1.SearchDeptName( deptCode );
00101:   RetriveOrgName( orgCode );
00102:   with DataModule1.CAT_Query do
00103:   begin
00104:     Close;
00105:     SQL.Clear;
00106:     SQL.Add( 'SELECT *' );
00107:     SQL.Add( 'FROM' );
00108:     SQL.Add( '  Transaction_Cat_Table' );
00109:     SQL.Add( 'WHERE' );
00110:     SQL.Add( '  Month = '+IntToStr( curSelMonth+1 ) );
00111:     SQL.Add( 'AND' );
00112:     SQL.Add( '  TelNo = '+#39+telId+#39 );
00113:     SQL.Add( 'AND' );
00114:     SQL.Add( '  Year = '+IntToStr( curYear ) );
00115:     SQL.Add( 'AND' );
00116:     SQL.Add( '  Period = '+IntToStr( curPeriod ) );
00117:     Open;
00118:   end;
00119:   if ( DataModule1.CAT_Query.IsEmpty ) then
00120:   begin
00121:     trans_Type:=INSERT_TYPE; DataModule1.CAT_Query.Insert;
00122:   end else
00123:   begin
00124:     trans_Type:=EDIT_TYPE; DataModule1.CAT_Query.Edit;
00125:   end;
00126: end;
00127: //---------------------------------------------------
00128: procedure TFitem2_2.ApplyUpdate;
00129: begin
00130:   with DataModule1.CAT_Query do
00131:   begin
00132:     DataModule1.Database1.StartTransaction;
00133:     try
00134:       ApplyUpdates;
00135:       DataModule1.Database1.Commit;
00136:     except
00137:       DataModule1.Database1.Rollback;
00138:     raise;
00139:     end;
00140:     commitUpdates;
00141:   end;
00142: end;
00143: procedure TFitem2_2.CalVAT_TAX( serviceCharge : double );
00144: var VATrate : double;
00145:     VATvalue, totalAmount : double;
00146:     deptCode : integer;
00147: begin
00148:   // get VAT rate
```

→

Delphi

```
00149:     VATrate:=DataModule1.Org_DS.DataSet.FieldByname( 'VAT_Rate' )AsFloat;
00150:    //Calculated VAT value & Total amount
00151:    VATValue:=( serviceCharge * VATrate ) / 100;
00152:    totalAmount:=serviceCharge + VATvalue;
00153:    with DataModule1.CAT_Query do
00154:    begin
00155:      FieldByName( 'Month' ).asInteger:=curSelMonth+1;
00156:      FieldByName( 'Year' ).asInteger:=curYear;
00157:      FieldByName( 'Period' ).asInteger:=curPeriod;
00158:      FieldByName( 'TelNo' ).asString:=Edit1.text;
00159:        DeptCode:=DataModule1.Dept_Query.FieldByName('Dept_No').AsInteger;
00160:      FieldByName( 'Dept_No' ).AsInteger:=DeptCode;
00161:      FieldByName( 'RecDate' ).asDateTime:=Date;
00162:      FieldByName( 'ServiceCharge' ).AsFloat:=serviceCharge;
00163:     FieldByName( 'VAT' ).AsFloat:=VATvalue;
00164:      FieldByName( 'TotAmount' ).AsFloat:=totalAmount;
00165:    end;
00166:    ApplyUpdate;
00167:    trans_Type:=NOT_SELECT;
00168:  end;
00169: //---------------------------------------------------
00170:  procedure TFitem2_2.InitVar;
00171:  begin
00172:    trans_Type:=NOT_SELECT;
00173:    PmonthlySelect.caption:='  ทำรายการของเดือน '+GetSelectMonth;
00174:    case curPeriod of
00175:      0 : Pperiod.caption:=' งวดที่ 1 : วันที่ 1 ถึง 15';
00176:      1 : Pperiod.caption:=' งวดที่ 2 : วันที่ 16 ถึง 30';
00177:    end;
00178:    RetrivePhoneNumber;
00179:  end;
00180:  procedure TFitem2_2.ReleaseVar;
00181:  begin
00182:    Fitem2_2.release; Fitem2_2:=nil;
00183:  end;
00184: //---------------------------------------------------
00185:  procedure TFitem2_2.FormCreate(Sender: TObject);
00186:  begin
00187:    InitVar;
00188:  end;
00189:  procedure TFitem2_2.FormCloseQuery(Sender: TObject; var CanClose: Boolean);
00190:  begin
00191:    ReleaseVar;
00192:    Canclose:=true;
00193:  end;
00194:  procedure TFitem2_2.BitBtn2Click(Sender: TObject);
00195:  begin
00196:    FPrMenu.Show; Fitem2_2.close;
00197:  end;
00198:  procedure TFitem2_2.DBGrid1CellClick(Column: TColumn);
00199:  begin
00200:     Edit1.text:=Column.Grid.SelectedField.AsString;
00201:    SearchTransaction( Edit1.text );
00202:    Edit1.Setfocus;
00203:  end;
00204:  procedure TFitem2_2.DBGrid1ColEnter(Sender: TObject);
00205:  begin
00206:     Edit1.text:=DBGrid1.Columns.Grid.SelectedField.AsString;
00207:    SearchTransaction( Edit1.text );
00208:    Edit1.SetFocus;
00209:  end;
00210:  procedure TFitem2_2.DBGrid1KeyUp(Sender: TObject; var Key: Word;
00211:    Shift: TShiftState);
00212:  begin
00213:     Edit1.text:=DBGrid1.Columns.Grid.SelectedField.AsString;
00214:  end;
00215:  procedure TFitem2_2.Edit1KeyUp(Sender: TObject; var Key: Word;
00216:    Shift: TShiftState);
00217:  var telNo : ansiString;
00218:  begin
00219:    case key of
00220:      VK_RETURN :
00221:      begin
00222:        telNo:=Edit1.text;
00223:  if (not DataModule1.Tel_Query.Locate( 'Tel_No',telNo,[ loCaseInsensitive ]))then
00224:        begin
00225:          Edit1.text:='';
00226:          ShowMessage( 'ไม่พบ หมายเลขโทรศัพท์ '+telNo+' ในระบบ' ); exit;
00227:        end;
00228:        SearchTransaction( Edit1.text );
00229:        Application.ProcessMessages;
00230:        DBEdit6.SetFocus; DBEdit6.SelectAll;
00231:      end;
00232:      VK_UP :
00233:      begin
00234:        DataModule1.Tel_Query.Prior;
00235:        Edit1.text:=DBGrid1.SelectedField.AsString;
00236:        SearchTransaction( Edit1.text );
00237:      end;
00238:      VK_DOWN :
00239:      begin
00240:        DataModule1.Tel_Query.Next;
00241:        Edit1.text:=DBGrid1.SelectedField.AsString;
00242:        SearchTransaction( Edit1.text );
00243:      end;
00244:    end;
00245:  end;
00246:  procedure TFitem2_2.DBEdit6KeyUp(Sender: TObject; var Key: Word;
00247:    Shift: TShiftState);
00248:  var serviceCharge : double; code : integer;
00249:  begin
00250:    case key of
00251:      VK_RETURN :
00252:      begin
00253:        if ( Edit1.text='' ) then exit;
00254:        val( DBEdit6.text, serviceCharge, code );
00255:        if ( code<>0 ) then
00256:        begin
00257:          ShowMessage( 'Invalid total amount' );
00258:          DBEdit6.SetFocus; DBEdit6.SelectAll;
00259:          exit;
00260:        end;
00261:        // in case, customer re-edit ( correct ) data
00262:        if ( trans_Type=NOT_SELECT ) then
00263:        begin
00264:          SearchTransaction( Edit1.text );
00265:          CalVAT_TAX( serviceCharge );
00266:          SearchTransaction( Edit1.text );
00267:        end
00268:        else CalVAT_TAX( serviceCharge );
00269:        Edit1.setfocus;
00270:      end;
00271:      VK_UP, VK_DOWN :
00272:      begin
00273:        Edit1.SetFocus; Edit1.SelectAll;
00274:      end;
00275:    end;
00276:  end;
00277:  procedure TFitem2_2.Button1Click(Sender: TObject);
00278:  var ret : word;
00279:  begin
00280:    ret:=MessageDlg( 'Delete '+Edit1.text+', are you sure ?', mtConfirmation,
00281:                     mbOKcancel, 0 );
00282:    if ( ret=mrOK ) then
00283:    begin
00284:      trans_Type:=DELETE_TYPE;
00285:      DataModule1.CAT_Query.Delete;
00286:      Application.ProcessMessages;
00287:      ApplyUpdate;
00288:      SearchTransaction( Edit1.text );
00289:    end;
00290:  end;
00291:  end.
```

Delphi



รูปที่ 5
ฟอร์ม FselMnth
สำหรับเลือกเดือน
ที่ต้องการทำรายการ

รูปที่ 6
ฟอร์ม FselJob
สำหรับเลือกชนิดรายการ
ที่ต้องการบันทึก

หน้าต่างบันทึกค่าบริการ การสื่อสาร (Fitem2_2/UITEM2_2.PAS)

ฟอร์ม Fitem2_2 แสดงดังรูปที่ 4 และมีส่วนของโปรแกรมแสดงอยู่ในลิสติงที่ 3 มีลักษณะการใช้ Query ในการ Retrive และ Update ข้อมูลคล้ายกับฟอร์ม Fitem1_2 (หน้าต่างบันทึกค่าบริการโทรศัพท์) ทุกประการ ต่างกันที่รูปแบบการใช้งาน และการใช้นำเสนอข้อมูลด้วย Data Control จุดประสงค์ที่ผมใช้วิธีการนำเสนอที่หลากหลาย เพื่อแสดงวิธีการอ่าน (Retrive) ข้อมูลในรูปแบบต่าง ๆ เนื่องจากในโลกของการทำงานจริง ๆ อาจต้องมีการใช้วิธีการต่าง ๆ ในการอ่านข้อมูลเพื่อให้ประสิทธิภาพในการดำเนินงานสูงสุด

การนำเสนอข้อมูลที่ค้นหาได้ในฟอร์ม Fitem1_2 จะใช้วิธีอ่านโดยตรงจาก Dataset เช่น

```
ftmp:=DataModule1.Trans_Query.FieldByName
( 'TotAmount' ).AsFloat;
Edit2.text:=Format('%8.2f', [ftmp]);
```

จะเป็นการอ่านข้อมูลโดยตรงจาก Org_DS Dataset ให้กับตัวแปร fTmp ที่มีแบบข้อมูลเดียวกัน แล้วจึงเปลี่ยนแปลง (Convert) ให้เป็นข้อมูลชนิดสตริงให้กับ Edit2.txt แตกต่างกันกับฟอร์มนี้ (Fitem2_2) ที่ใช้ Data Controls (TDBEdit) ในการเข้าถึงข้อมูล เช่น

กำหนดให้ DBEdit6 ใช้รับข้อมูลแทน Tedit จะต้องกำหนด Properties ของ DBEdit6 ให้เชื่อมโยงไปยัง Dataset ดังนี้

```
DBEdit6.DataSource:=' DataModule1.CAT_DS';
// กำหนดให้ เชื่อมโยงไปที่ CAT_DS Dataset
```

และ

```
DBEdit6.DataField:=' ServiceCharge';
// กำหนดให้ DBEdit6 อ้างถึงฟิลด์ 'SrviceCharge'
```

หน้าต่างเลือกเดือน (FselMnth/USELMNTH.PAS)

ฟอร์ม FselMnth แสดงดังรูปที่ 5 เป็นฟอร์มสำหรับเลือกเดือนที่ต้องการบันทึกรายการค่าบริการ โดยอ่านค่าจาก RadioGroup1.ItemIndex แล้วระบุให้ค่ากับตัวแปร curSelMonth เพื่อนำไปในประกอบกับ Query สำหรับการค้นหาข้อมูล และบันทึกข้อมูล ในฟอร์มต่อไป

หน้าต่างเลือกชนิดรายการ (FselJob/USELJOB.PAS)

ฟอร์ม FselJob แสดงดังรูปที่ 6 ทำหน้าที่สำหรับเลือกชนิดรายการคือ บันทึกค่าบริการโทรศัพท์ และค่าบริการของการสื่อสารฯ โดยจะกำหนดค่าให้กับตัวแปร curSelJob และเลือกรันฟอร์ม FITEM1_1 สำหรับการบันทึกค่าบริการแบบแรก และฟอร์ม FITEM2_1 สำหรับการบันทึกค่าบริการอื่น

เป็นอย่างไรบ้างครับ สำหรับเนื้อหาในตอนที่ 2 ฉบับนี้ ซึ่งจะครอบคลุมในส่วนของ Query เกือบทั้งหมด ผมยังคงเหลือฟอร์มที่เกี่ยวกับการแก้ไขข้อมูลด้วย Query อีก 3 ฟอร์มที่ยังไม่ได้กล่าวถึง คือ ฟอร์มแก้ไขเพิ่มเติมหมายเลขโทรศัพท์ (FadjTel) ฟอร์มแก้ไขเพิ่มเติมรายชื่อฝ่ายในบริษัท (FadjDept) และ ฟอร์มแก้ไของค์กร/บริษัท ที่ให้บริการโทรศัพท์ (FadjDept) ผมขอเก็บไว้คุยพร้อมกับส่วนการสร้าง Report ในฉบับหน้า ซึ่งเป็นตอนจบของบทความชุดนี้ สำหรับการสั่งซื้อซอร์สโค้ดของโปรแกรม ผมได้จัดทำไว้ใน Vol. 82 ของไมโครคอมพิวเตอร์ฉบับที่ 167 แล้ว หากท่านไม่ประสงค์จะซื้อ Vol. 82 ผมก็ได้จัดทำไว้ใน Vol. 83 ของฉบับนี้ด้วยครับ ฉบับนี้คงต้องจากกันเท่านี้ก่อน ฉบับหน้าพบกันใหม่ สวัสดีครับ